Preface: Special section on Graph Inspection and Traversal Engineering (GRAPHITE 2014)

Bosnacki, Dragan; Edelkamp, Stefan; Lluch Lafuente, Alberto; Wijs, Anton

Published in:
Science of Computer Programming

Link to article, DOI:
10.1016/j.scico.2016.06.007

Publication date:
2016

Document Version
Peer reviewed version

Special Issue on Graph Inspection and Traversal Engineering (GRAPHITE 2014)

Dragan Bošnački[a], Stefan Edelkamp[b], Alberto Lluch Lafuente[c], Anton Wijs[d]

[a] *Eindhoven University of Technology, The Netherlands*
[b] *University of Bremen, Germany*
[c] *Technical University of Denmark, Denmark*
[d] *Eindhoven University of Technology, The Netherlands*

This special issue presents extended versions of a selection of papers presented at the 3rd International Workshop on GRAPH Inspection and Traversal Engineering (GRAPHITE 2014), which took place on April 5, 2014 in Grenoble, France, as a satellite event of the 17th European Joint Conferences on Theory and Practice of Software (ETAPS 2014).

The aim of GRAPHITE is to foster the convergence on research interests from several communities dealing with graph analysis in all its forms in computer science, with a particular attention to software development and analysis. Graphs are used to represent data and processes in many application areas, and they are subjected to various computational algorithms in order to analyse them. Just restricting the attention to the analysis of software, graph analysis algorithms are used, for instance, to verify properties using model checking techniques that explore the system's state space graph or static analysis techniques based on control flow graphs. Further application domains include games, planning, and network analysis. Very often, graph problems and their algorithmic solutions have common characteristics, independent of their application domain. The goal of GRAPHITE is to gather scientists from different communities, who do research on graph analysis algorithms, such that awareness of each others' work is increased.

The authors of some of the most interesting and promising papers presented at GRAPHITE 2014 were invited to submit an extended version of their work to this special issue. The articles presented here were thoroughly reviewed by competent reviewers, including some reviewers involved in the selection of the original workshop papers and new reviewers that provided a

fresh perspective. The review process has guaranteed that the papers have been significantly extended and improved with respect to the original workshop versions.

This special issue presents two articles:

- *Specifying and Executing Optimizations for Generalized Control Flow Graphs*, by William Mansky, Elsa L. Gunter, Dennis Griffith, and Michael D. Adams, presents Morpheus, a domain-specific language for the specification of program transformations as rewrites on control flow graphs with temporal logic side conditions. The article is illustrated with a set of typical compiler optimisations of sequential and parallel programs.

- *Using Graph Distances for Named-Entity Linking*, by Roi Blanco, Paolo Boldi, and Andrea Marino studies a new variant of the Maximum Capacity Representative Set graph problem, with a focus on its application to entity-linking, a fundamental task to automatically associate text portions to semantic concepts. The article shows that the problem is NP-hard for general graphs and proposes several heuristics that are evaluated on real-world datasets extracted from Wikipedia.

We are grateful to all authors for their great contributions to the special issue. We would like to thank as well the reviewers of the articles for their effort to help the authors improve their articles, and all the members of the Program Committee of GRAPHITE, who helped us in the selection and improvement of the original workshop papers. Finally, we would also like to thank the editors of *Science of Computer Programming* for their patience and support during the preparation of the special issue.